Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное

образовательное учреждение высшего образования

Санкт-Петербургский университет аэрокосмического приборостроения

ГУАП

КАФЕДРА № 2

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Кандидат тех. Наук, доцент |  |  |  | С.Л Козенко |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №4  **“Численное интегрирование”** |
| по дисциплине: Вычислительная математика |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. | 4134к |  |  |  | Костяков НА |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург

2023

Цели работы: а) Составить схему алгоритма и программу на языке C/C++ решения задачи по теме «Численное интегрирование» в соответствии с индивидуальным заданием (варианты заданий приведены ниже – табл. 1).



# Математическая часть

***2.1. Постановка задачи***

##### Пусть требуется найти определенный интеграл

 (2.1)

Где функция *f(x)* непрерывна на отрезке *[a,b]*. Если функция *f(x)* задана формулой и мы умеем найти неопределенный интеграл *F(x)*, то определенный интеграл вычисляется по формуле

 (2.2)

(формула Ньютона-Лейбница).

Если же неопределенный интеграл данной функции мы найти не умеем, или по какой-либо причине не хотим воспользоваться формулой (2.2), или если функция *f(x)* задана графически или таблицей, то для вычисления определенного интеграла применяют приближенные формулы. Для приближенного вычисления интеграла (2.1) существует много численных методов, из которых выделим три:

1) метод прямоугольников;

2) метод трапеций;

3) метод Симпсона (парабол).

При вычислении интеграла следует помнить, каков геометрический смысл определенного интеграла. Если *f(x)≥0* на отрезке *[a,b],* то



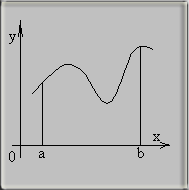
численно равен площади фигуры, ограниченной графиком функции *y=f(x),* отрезком оси абсцисс, прямой *x=a* и прямой *x=b* (рис.2.1).

Рис.2.1 Геометрический смысл численного интеграла

Таким образом, вычисление интеграла равносильно вычислению площади криволинейной трапеции.

***2.2. Метод прямоугольников***

Разделим отрезок *[a,b]* на *n* равных частей, т.е. на *n* элементарных отрезков. Длина каждого элементарного отрезка *h=(b-a)/n*. Точки деления будут: *x0=a, x1=a+h, x2=a+2h, … , xn-1=a+(n-1)h,* *xn=b*. Эти числа будем называть узлами. Вычислим значения функции *f(x)* в узлах, обозначим их *y0, y1, y2, …, yn*. Стало быть, *y0=f(a), y1=f(x1), …, yn=f(b).* Числа *y0, y1, y2, …, yn* суть ординаты точек графика функции, соответствующих абсциссам *x0, x1, x2, …, xn* (рис.2.2). Из рис.2.2 следует, что площадь криволинейной трапеции приближенно заменяется площадью многоугольника, составленного из *n* прямоугольников. Таким образом вычисление определенного интеграла сводится к нахождению суммы *n* элементарных прямоугольников.

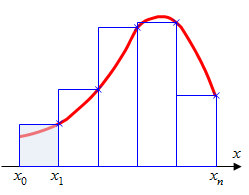
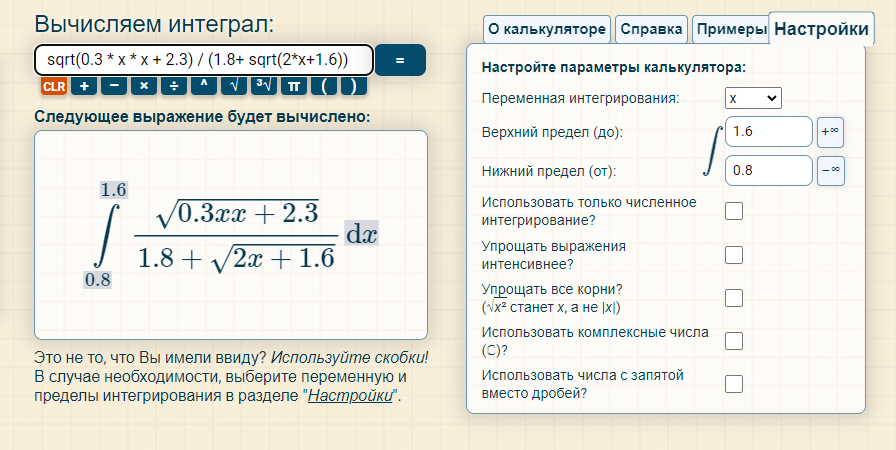
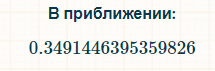


Рис. 2.2. Геометрическая иллюстрация метода правых прямоугольников

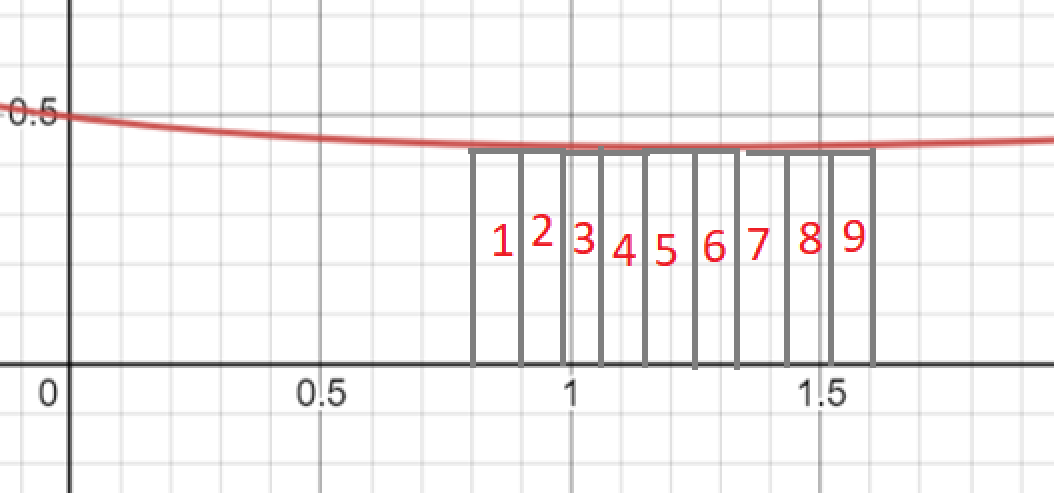


# Аналитические расчеты

Проверка интеграла с помощью калькулятора 

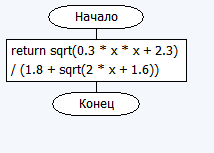


График

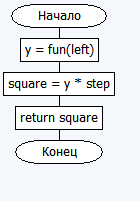


# Блок Схема

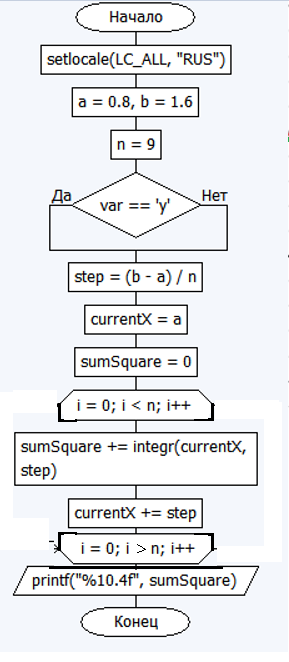
Double fun();



Double Integr();



Main()



# Листинг программы

#include <iostream>

#include <cmath>

double fun(double x) {

return sqrt(0.3 \* x \* x + 2.3) / (1.8+ sqrt(2\*x+1.6));

}

double integr(double left, double step) {

double y = fun(left);

double square = y \* step;

return square;

}

int main()

{

setlocale(LC\_ALL, "RUS");

double a = 0.8, b = 1.6;

int n = 9;

double step = (b - a) / n;

double currentX = a;

double sumSquare = 0;

for (int i = 0; i < n; i++) {

sumSquare += integr(currentX, step);

currentX += step;

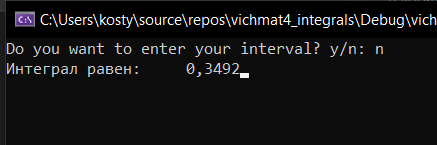
}

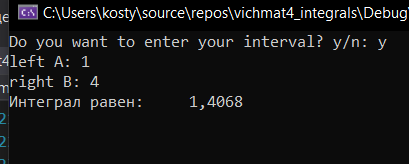
std::cout << "Интеграл равен: ";

printf( "%10.4f", sumSquare );

}

# Результат работы программы





# Сравнение результатов аналитических расчетов с программными

Аналитическая часть 0.3491

Программная 0.3492

Результаты различаются на 0.0001

# Вывод

Я освоил методы интегрирования и их реализацию на языке c++